

## BAB III

### METODOLOGI PENELITIAN

#### 3.1 Alat dan Bahan Penelitian

##### 3.1.1 Alat Penelitian

Alat Penelitian dalam penelitian ini adalah dengan menggunakan satu buah Laptop dengan perangkat lunak *Compiler* NetBeans IDE 8.2, Notepad++, *Xampp Control Panel* v3.2.1, *Framework* CodeIgniter, BarTender Seagull Scientific, Artisteer, UC Browser, Mozilla Firefox, Opera, dengan spesifikasi Laptop sebagai berikut:

- 1) OS Microsoft Windows 10 LTSB 2015 Enterprise 64-bit
- 2) Processor Intel® Pentium® CPU N3540 @ 2.16GHz (4 CPUs),  
~2.2GHz
- 3) Memory 4096MB RAM
- 4) Barcode Scanner Printpos

##### 3.1.2 Bahan Penelitian

Bahan materi yang diperlukan dalam penelitian ini diuraikan sebagai berikut:

- 1) Makalah atau Jurnal yang berkaitan dengan Sistem Informasi berbasis Web, *Framework* CodeIgniter, PHP MySQL, model *waterfall*, Rekayasa Perangkat Lunak.

2) Data yang dibutuhkan dalam penelitian ini mencakup data Sumber informasi dalam analisis kebutuhan adalah laboran dari laboratorium di Program Studi Teknik Elektro FT UMY. Obyek penelitian meliputi aspek *functionality, reliability, efficiency, portability, dan maintainability* adalah Sistem Informasi Inventaris Laboratorium dengan bantuan ahli aplikasi *web, tools* penguji, dan pengembang sendiri; sedangkan subyek penelitian dalam aspek *usability* adalah teknisi laboratorium dan mahasiswa Program Studi Teknik Elektro FT UMY yang berinteraksi dengan laboratorium.

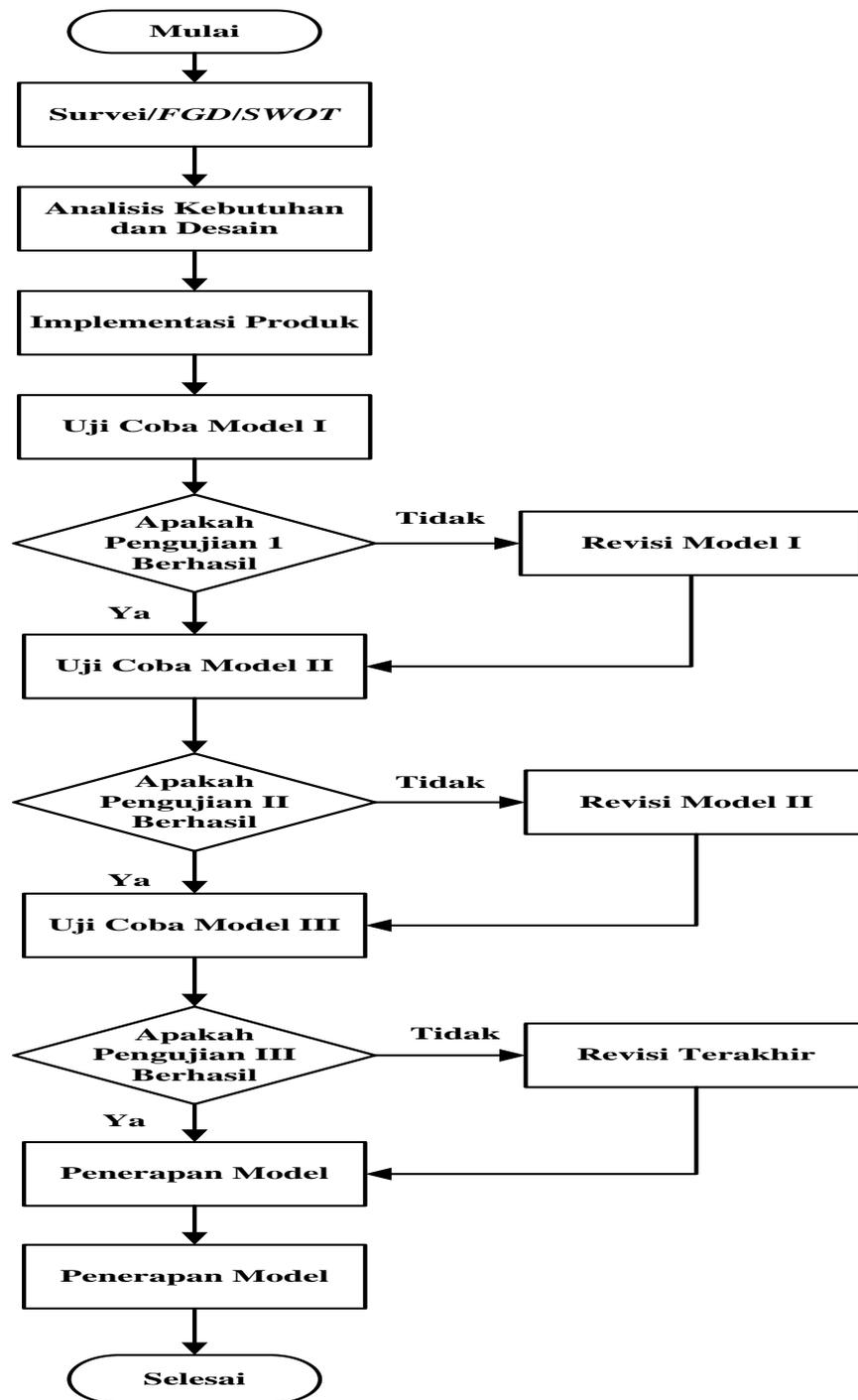
### **3.2 Lokasi Penelitian**

Adapun Penelitian sebagai dasar dalam perencanaan dan perancangan penelitian dilaksanakan di Laboratorium Program Studi Teknik Elektro FT UMY.

### **3.3 Model Penelitian**

Penelitian ini menggunakan metode riset dan pengembangan (*Research and Development*) dan menggunakan model pengembangan *waterfall*.

Langkah-langkah penelitian dan pengembangan yang dilakukan dalam penelitian ini mengacu pada tahap penelitian Borg dan Gall, seperti bagan versi Endang Mulyatiningsih (2011, hal. 149), dengan penyesuaian, dalam Gambar 3.1 berikut.



Gambar 3.1 Desain *flowchart* penelitian

### **3.3.1 Analisis Kebutuhan**

Tahap ini merupakan tahap awal yang berisi pengumpulan data yang dimulai dari adanya observasi yang dilakukan peneliti untuk melihat masalah yang ada di lapangan. Kegiatan dalam tahap ini meliputi observasi dan wawancara. Observasi adalah melakukan pengamatan secara langsung ke obyek penelitian untuk melihat dari dekat kegiatan yang dilakukan (Guritno, Sudaryono, & Rahardja, 2011, hal. 134). Wawancara adalah suatu cara pengumpulan data yang digunakan untuk memperoleh informasi langsung dari sumbernya (Guritno, Sudaryono, & Rahardja, 2011, hal.131). Dari wawancara dan observasi yang dilakukan, dapat ditentukan tujuan, syarat, dan spesifikasi sistem yang dibutuhkan.

### **3.3.2 Desain**

Proses desain meliputi serangkaian langkah untuk menggambarkan semua aspek perangkat lunak yang dibangun. Aspek yang dihasilkan dalam tahap ini antara lain adalah representasi data, arsitektur, *interface*, dan prosedur (Pressman, 2002, hal. 426). Penjelasan untuk tiap desain adalah sebagai berikut:

#### **3.3.2.1 Desain Basis Data**

Desain basis data didasarkan dari kebutuhan sistem dan bagaimana hubungan antar data yang didasarkan pada informasi syarat dan spesifikasi sistem. Desain pada tahap ini akan diimplementasikan pada server Sistem Informasi Inventaris Laboratorium tersebut.

### 3.3.2.2 Desain Arsitektur

Sasaran utama desain arsitektur adalah untuk mengembangkan struktur program *modular* dan merepresentasikan hubungan kontrol antar modul (Pressman, 2002, hal. 438). Desain arsitektur ini menggunakan notasi *Unified Modelling Language* (UML), yang meliputi beberapa diagram di bawah ini.

#### 1) *Use Case Diagram*

*Use case diagram* terdiri dari *actor* dan aksi yang dapat dilakukannya. Pada pengembangan web Laboratorium ini, *use case diagram* menjelaskan hubungan antara sistem dengan *administrator* dan pengguna.

#### 2) *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem Laboratorium dari kelas-kelas yang akan dibuat dalam membangun sistem ini. *Class diagram* terdiri dari nama kelas, atribut, dan operasi yang ada di dalamnya.

### 3.3.2.3 Desain Interface

Berdasarkan desain arsitektur yang telah terbentuk, desain *interface* dibuat meliputi dua tampilan, yaitu untuk tampilan *admin* dan tampilan *user* biasa. Desain yang dibuat menyesuaikan fungsi dari jenis pengguna.

#### 1) Desain *InterfaceAdmin* (*Dashboard*)

Desain *interfaceadmin* meliputi rancangan tampilan Laboratorium yang diakses oleh admin.

#### 2) Desain *Interface User* Biasa

Desain interface user meliputi rancangan tampilan Laboratorium yang diakses oleh *user* biasa.

### 3.3.2.4 Desain Prosedural

Desain Prosedural terjadi setelah data, desain arsitektur, dan *interface*, dibangun. Dalam dunia yang ideal, spesifikasi prosedural diperlukan untuk menetapkan detail algoritma yang akan dinyatakan dalam suatu bahasa ibu seperti bahasa inggris (Pressman, 2002, hal. 475). Pada pembuatan desain prosedural ini dapat menggunakan bagan alir (*flowchart*) (Pressman, 2002).

### 3.3.3 Implementasi

Tahap Implementasi berupa tahapan merealisasikan desain yang telah ada sehingga terbentuk sistem program Laboratorium yang siap digunakan. Proses implementasi yang dilakukan dalam pengembangan ini adalah berbasis *web* PHP menggunakan *Framework* CodeIgniter.

### 3.3.4 Pengujian

- 1) Tahap Pengujian Unit: dilakukan *white-box testing* dengan menguji tiap modul yang bekerja dalam sistem.

- 2) Tahap Pengujian Integrasi: dilakukan dengan menelusuri dua atau lebih unit yang saling berhubungan apakah sudah sesuai dengan rencana kerja.
- 3) Tahap Pengujian Sistem: dilakukan *black-box testing* dengan melakukan uji *checklist* tiap fungsi pada sistem yang dilakukan oleh ahli.
- 4) Tahap pengujian Penerimaan: dilakukan dengan uji kuesioner pada *user*.

### **3.4 Metode dan Alat Pengumpulan Data**

Teknik pengumpulan data yang dilakukan pada penelitian ini meliputi:

#### **3.4.1 Observasi**

Teknik observasi dilakukan untuk mengumpulkan data pada aspek *functionality, reliability, efficiency, portability, dan maintainability*.

##### 1) Aspek *Functionality*

Berdasarkan definisi karakter dan subkarakteristik aspek *functionality*, pengujian dilakukan dengan *checklist* daftar fungsi untuk melihat kesesuaian kesatuan fungsi dalam melakukan tugas tertentu. Fungsi yang diuji disesuaikan dengan kebutuhan *user (user requirement list)*. Pengujian ini dilakukan dengan menggunakan 3 responden, sebagai ahli di bidang aplikasi *web*.

Berdasarkan *requirement list* yang didapat, *instrument* pengujian *functionality* yang digunakan dalam penelitian ini yakni pada Tabel 3.1 berikut.

1) Halaman *Administrator (Back-End)*

**Tabel 3.1** Instrumen pengujian *functionality* halaman admin

No.	Fungsi	Pertanyaan	Ya	Tidak
1	<i>Login</i>	Apakah fungsi <i>login</i> dapat berfungsi dengan benar?		
2	Menu	Apakah menu navigasi utama dapat berfungsi dengan benar?		
3	Manajemen	Apakah <i>datauser</i> dapat tampil dengan benar?		
4	<i>User</i>	Apakah fungsi menambah <i>user</i> baru dapat berfungsi dengan benar?		
5		Apakah fungsi mengubah data <i>user</i> dapat berfungsi dengan benar?		
6		Apakah fungsi untuk menghapus data <i>user</i> dapat berfungsi dengan benar?		
7		Apakah fungsi untuk mencetak data <i>user</i> berfungsi dengan benar?		
No	Fungsi	Pertanyaan	Ya	Tidak
8	Manajemen Perangkat	Apakah data perangkat dapat tampil dengan benar?		
9		Apakah fungsi menambah perangkat baru		

		berfungsi dengan benar		
10		Apakah fungsi mengubah data perangkat dapat berfungsi dengan benar?		
11		Apakah fungsi untuk menghapus data perangkat dapat berfungsi dengan benar?		
12		Apakah fungsi untuk mencetak data perangkat berfungsi dengan benar?		
13		Apakah fungsi penyimpanan otomatis akhir masa manfaat barang, ketika dilakukan penyimpanan barang baru, dapat berfungsi dengan benar?		
14		Apakah fungsi untuk menampilkan perangkat yang mencapai akhir masa manfaat dapat berfungsi dengan benar?		
<b>No</b>	<b>Fungsi</b>	<b>Pertanyaan</b>	<b>Ya</b>	<b>Tidak</b>
15	Antrian	Apakah data semua antrian dapat tampil dengan benar?		
16		Apakah fungsi mencari data perangkat saat menambah antrian dapat berfungsi dengan benar?		
17		Apakah fungsi menambah antrian dapat berfungsi dengan benar?		
18		Apakah fungsi untuk memproses data antrian (memindahkan ke tabel peminjaman) dapat berfungsi dengan benar?		
19		Apakah fungsi untuk mencetak data antrian berfungsi dengan benar?		

20	Peminjaman	Apakah data peminjaman dapat tampil dengan benar?
21		Apakah fungsi menambah peminjaman baru dapat berfungsi dengan benar?
22		Apakah fungsi mengubah data peminjaman dapat berfungsi dengan benar?
23		Apakah fungsi untuk menghapus data peminjaman dapat berfungsi dengan benar?
24		Apakah fungsi untuk mencetak data peminjaman berfungsi dengan benar?
25		Apakah fungsi untuk selesai peminjaman berfungsi dengan benar (status barang menjadi 'tersedia')?
26	<i>Logout</i>	Apakah fungsi <i>logout</i> dapat berfungsi dengan benar?

**Tabel 3.2** Instrumen Pengujian *Functionality* Halaman *User*

No.	Fungsi	Pertanyaan	Ya	Tidak
1.	<i>Login</i>	Apakah fungsi <i>login</i> dapat berfungsi dengan benar?		
2.	Navigasi Utama	Apakah menu navigasi utama dapat berfungsi dengan benar?		

3. Edit Profil Apakah data *user* dapat tampil dengan benar?
4. Apakah fungsi untuk mengubah data profil dapat berfungsi dengan benar?
5. Pemesanan Apakah data perangkat yang disediakan dapat tampil dengan benar?
6. Apakah fungsi proses pencairan data perangkat sudah berfungsi dengan benar?
7. Apakah fungsi untuk memesan perangkat dapat berfungsi dengan benar?
8. Antrian Apakah data antrian dapat tampil dengan benar?
9. *Logout* Apakah fungsi *logout* dapat berfungsi dengan benar?

## 2) Aspek *Reliability*

Untuk menguji kemampuan perangkat lunak dalam mempertahankan kinerja, dilakukan dengan memberikan *stress testing* yang menjadikan banyak *user* dan banyak aksi dimasukkan pada *website*. Dengan pengujian ini, dapat diketahui pada tingkat mana sistem dapat meng-*handle* kinerja. Pengujian dilakukan dengan bantuan LoadImpact dan Webserver Stress Tool 7.

### 3) Pengujian *Efficiency*

Pengujian pada aspek ini dilakukan dengan mencatat *load time* dari halaman-halaman di *website* Laboratorium. Pengujian dilakukan dengan bantuan *tools* YSlow dan GTmetrix.

### 4) Pengujian *Portability*

Pengujian dilakukan dengan uji lapangan dengan alat yang berbeda, serta menggunakan bantuan *website* yang mampu memberikan simulasi berjalannya sistem di berbagai lingkungan. *Website* yang digunakan antara lain adalah [webpagetest.org](http://webpagetest.org) dan diuji langsung pada berbagai macam *web browser*, baik versi *desktop* maupun *mobile*.

### 5) Pengujian *Maintainability*

Pengujian pada aspek ini menggunakan ukuran yang diuji oleh peneliti langsung di lapangan secara operasional (Land, 2002). Sesuai instrumen pengujian *maintainability* Land, pengujian ini meliputi 3 aspek, yaitu *instrumentation*, *consistency*, dan *simplicity*. Keterangan penilaian aspek ini tertuang pada Tabel 3.3 berikut.

**Tabel 3.3** Instrumen Uji *Maintainability*

Aspek	Aspek yang dinilai	Hasil yang diperoleh
<i>Instrumentation</i>	Terdapat peringatan dari sistem jika terjadi kesalahan beserta identifikasi kesalahan	Ketika ada kesalahan yang dilakukan oleh user, maka sistem akan mengeluarkan peringatan untuk

		mengidentifikasi kesalahan.
<i>Consistency</i>	Penggunaan satu model rancangan pada seluruh rancangan sistem	Bentuk rancangan sistem pengolah data mempunyai satu bentuk yang sama. Hal ini dapat dilihat pada bagian implementasi sistem.
<i>Simplicity</i>	Kemudahan dalam pengelolaan, perbaikan, dan pengembangan sistem	Mudah untuk dikelola, diperbaiki, dan dikembangkan. Hal ini dapat dilihat pada tahapan proses penulisan kode program.

### 3.4.2 Kuesioner

Teknik kuesioner digunakan untuk mengumpulkan data pada aspek *usability*. Instrumen yang digunakan adalah angket *usability The Standardized Universal Percentile Rank Questionner* (SUPR-Q) yang dirilis oleh Jeff Sauro, seperti dalam bukunya *Quantifying The User Experience Practical Statistics for User Research* (Sauro & Lewis, 2012).

## 3.5 Teknik Analisis Data

### 3.5.1 Aspek *Functionality*

Hasil *checklist* yang diperoleh dari beberapa ahli tersebut dilakukan perhitungan persentase dengan rumus sebagai berikut:

$$\text{Persentase } \textit{Functionality} = \frac{\text{Hasil yang diperoleh}}{\text{Hasil Maksimal}} \times 100\%$$

Hasil persentase diatas kemudian dibandingkan dengan skala likert, yaitu skala yang digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang suatu kejadian (Guritno, Sudaryono, & Rahardja, 2011, hal.110).Pengelompokan tingkat persentase sesuai skala likert yaitu pada Tabel 3.5 berikut.

**Tabel 3.5** Interpretasi Skor Skala Likert

<b>Persentase Skor</b>	<b>Keterangan</b>
0%-20%	Sangat Tidak Baik
20%-40%	Tidak Baik
40%-60%	Netral
60%-80%	Baik
<b>80%-100%</b>	<b>Sangat Baik</b>

### 3.5.2 Aspek *Reliability*

Analisis untuk aspek *reliability*, yang diuji dengan bantuan *Webserver Stress Tool* dan *Website LoadImpact* ([www.loadimpact.com](http://www.loadimpact.com)), akan dihasilkan nilai *reliability* sesuai model nelson.

$$R1 = 1 - \frac{ne}{n}$$

Keterangan: R1= nilai *reliability*

ne= jumlah input yang gagal

n= jumlah input

Hasil persentase tersebut dibandingkan dengan standar uji *reliability*, yaitu standar Telcordia. Standar Telcordia mengatakan bahwa hasil pengujian dikatakan memenuhi aspek *reliability* jika persentase bernilai minimal 95% (Asthana & Olivieri, 2009).

### 3.5.3 Aspek *Usability*

Karakteristik kualitas *usability* dianalisis dengan menghitung persentase dari data kuesioner *The Standardized Universal Percentile Rank Questionner* (SUPR-Q) yang dirilis oleh Jeff Souro (Sauro & Lewis, 2002.). Catatan khusus untuk perhitungan pertanyaan nomor 13, diambil setengah ( $\frac{1}{2}$ ) dari nilai yang ada.

$$\text{Persentase Usability} = \frac{\text{Skor yang diperoleh}}{\text{Skor maksimal}} \times 100\%$$

Dimana,

Skor maksimal= jumlah item x nilai maksimal per item x  
jumlah responden  
Persentase yang didapat kemudian  
diinterpretasikan dengan skala likert.

### 3.5.4 Aspek *Efficiency*

Karakteristik kualitas *efficiency* didapat dari data *load time* dari pengujian menggunakan aplikasi YSlow dan GTMetrix. Rata-rata *load time* yang didapat dikatakan memenuhi uji aspek *efficiency* ketika 95%

*access request* kurang dari 4 detik, sesuai aturan pada GTMetrix sendiri (Gossamer Threads, 2014).

### **3.5.5 Aspek *Portability***

Analisis untuk aspek *portability* dilakukan dengan mencoba menjalankan aplikasi ini di berbagai *web browser*. Apabila percobaan berjalan dengan baik di semua *web browser* yang diujikan, maka *web* dinyatakan memenuhi uji aspek *portability*.

### **3.5.6 Aspek *Maintainability***

Karakteristik kualitas aspek *maintainability* diukur menggunakan *metrics* pengujian *maintainability* yang menghasilkan kondisi riil dari pengujian yang dilakukan secara operasional di lapangan. Jika *web* yang diuji lolos untuk semua aspek dalam instrumen maka *web* dinyatakan memenuhi uji aspek *maintainability*.